|  |  |
| --- | --- |
| Logo_FPT_University_doc |  |

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |

BUS USER SUPPORT SYSTEM

User Requirement Specification

|  |  |
| --- | --- |
| BUS USER SUPPORT SYSTEM | |
| **Group Members** | |  |  | | --- | --- | | Nguyễn Thành Nam | SE02942 | | Trịnh Thị Tuyết Mai | SE02630 | | Bùi Bích Phương | SE02704 | | Trần Tú Anh | SE02596 | |
| **Supervisor** | Mr. Nguyễn Văn Sang |
| **Project code** | BUSS |

- Hanoi, 06/2014 -

# Introduction

## Purposes

This document will provide the description of general requirements and non-functional requirements of BUSS system. Also, it does describe the requirement scope of each phase of the project.

## Definition and Acronyms

|  |  |
| --- | --- |
| Acronym & Abbreviation | Definition |
| BUSS | Bus User Support System |
| FU | FPT University |
| BU | Bus User |
| Q&A | Question and Answer |
|  |  |

**Table 1-1: Definition and Acronyms**

# Overall Description

## Business Process Overview

### Path finding workflow



Figure 1-1: Path Finding

**Brief description**: Users input the departure and destination location to find paths available for them to travel by bus. The results will automatically be sorted based on collected personal usage data and can be rearrange or filtered to match users' preferences:

**Step 1**: User inputs departure and destination location

**Step 2**: System will calculate and return several possible paths as result

**Step 3**: User choose a path of preference

**Step 4**: BUSS system will contact buses that contributing in this path to get location of the nearest one to your place

**Step 5**: BUSS give your detailed direction to get on buses and arrive at your preferred destination.

### Bus tracking workflow



Figure 1-2: Bus tracking

**Brief description**: Once users get on a bus, they can setup a tracker which will keep an eye on the bus all the time. The user only have to choose a stop they want to get off, and leave the rest to the tracker. Whenever the bus comes near that predefined stop, it will notify them to prepare the leave:

**Step 1**: Start the tracker and choose a preferred stop on a bus’s route

**Step 2**: System will mark the location of the chosen stop on the map and continuously contact the app to request for its current location

**Step 3**: The app will return its current location to server each time it’s been asked

**Step 4**: BUSS system will continuously compare user’s location with the marked location on the map. If the distance is less than 200m, it will notify user to get off the bus.

### Bus Reminder



Figure 1-3: Bus tracking

**Brief description**: the app functions similarly to an alarm yet with greater effort by integrating map information and bus information into users' schedule. To use this feature, the user first sets up a timeframe at about which they need to hit the road. User then chooses a departure location and a bus that they should get on. About time, the app will notify user if there are buses of the preferred kind which about to get to the nearest busstop of the departure location. This feature function well in combination with **Path finding** and **Bus tracking** features:

**Step 1**: the user sets up a timeframe at about which they need to hit the road, then chooses a departure location and a bus that they should get on

**Step 2**: About time, the system will contact buses to know if there is any bus of this kind is near the departure location.

**Step 3**: If there is, the app will notify user, and keep repeat from step 1 until user manually dismiss the reminder.

## Product Features

* **Search bus**

Allows users to search for specific bus through several scenarios and receive detail information about specific buses.

* **Find path**

Allows users to input the departure and destination location to find paths available for them to travel by bus

* **Manage reminder**

Allow users to add, view, edit and delete bus reminder.

* **Bus track**

Allows users set up tracker which will notify them when the bus is about to reach their preferred stop.

* **Interaction and Statistics**

Watch and analyze usage data to create a virtual interaction environment among users.

## User Characteristic

There are 2 kinds of user that BUSS supports:

* **Authenticated user**: user whose phone number was input and used as a synchronization token. Authorized user can sync and restore their usage data any time on any device using their phone number
* **Unauthenticated user**: user whose phone number was not input. Therefore, their usage data might be lost in several cases.

Both kinds of user can freely make use of all other features that the system offers.

# Functional Requirements

## Search bus

**Access right: All User**

This function allows users to search for specific bus through one of these scenarios:

* Look up from the bus list: Display a full list of all the buses available in the current city. User can filter this list by number or names of the streets they pass.
* Look for buses that pass or are near the current location of user.
* Look for buses that travel from the current location to a predefined destination.

## View bus detail

**Access right: All User**

This function allows users to view the bus route, bus stop, operation time and location of the nearest bus in comparison with user’s current location.

## Find path

**Access right: All User**

This function allows users to input the departure and destination location to find paths available for them to travel by bus. They can know about the location of the bus on the map, the time, the distance and then they can choose the most suitable bus for their demand.

## Manage reminder

**Access right: All User**

### Add reminder

This function allows users to choose the bus they want to travel, set up the departure, the recurrence and when this bus is coming to the departure, this application will remind user.

### View reminder

This function allows users to view all the reminders they added before. If this reminder is not necessary, user can turn off it or change it by tapping on it to see the reminder’s detailed setting page.

### Delete reminder

This function allows users to delete a reminder when it is not needed anymore.

## Manage favorite

**Access right: User**

### Add favorite

This function allows users to add some places or buses they care. When the user search place (or bus), the place (or bus) in favorite list will be prioritized in the suggest list.

### Delete favorite

This function allows users to delete the place or bus from favorite list when the user does not need it anymore.

## Bus Track

**Access right: User**

This function allows users set up tracker which will keep an eye on the bus all the time. User has to choose a stop they want to get off, and leave the rest to the tracker. Whenever the bus comes near that predefined stop, it will notify user to prepare the leave.

## Route navigate

**Access right: User**

This function gives users step-by-step detailed directions:

* How to get to the predefined destination mostly by bus.
* The users can know exactly the distance traveled by bus as well as the total distance.
* In addition, maps and information will be refresh each time a step completed, continuously guide users until they reach their destination.

# Non-functional Requirements

* Maximum respond time for offline function is 1 second. Maximum respond time for online function is 20 seconds.
* App functions quickly and precisely to ensure user’s satisfaction.
* GUI is simple, clear and easy to navigate among functions and takes less steps to use.
* Ensure data security capabilities.
* Support for multiple users at a same time, minimize overload situation.